CERTIFICATE OF MAILING via EXPRESS MAIL
37 C.F.R. § 1.10

PURSUANT TO 37 C.F.R. § 1.10, I HEREBY CERTIFY THAT I HAVE A REASONABLE BASIS FOR BELIEF THAT THIS CORRESPONDENCE IS BEING DEPOSITED WITH THE UNITED STATES POSTAL SERVICE AS EXPRESS MAIL POST OFFICE TO ADDRESSEE, ON THE DATE BELOW, IN AN ENVELOPE ADDRESSED TO:

MAIL STOP APPEAL BRIEF - PATENTS
HONORABLE COMMISSIONER FOR PATENTS
P.O. BOX 1450
ALEXANDRIA, VA 22313-1450

Ronald L. Chichester

RONALD L. CHICHESTER REG. NO. 36,765

JULY 7, 2004 — DATE
EV449865785US — EXPRESS MAIL LABEL

APPLICATION NUMBER:	09/892,837
FILING DATE:	JUNE 27, 2001
FIRST NAMED INVENTOR:	KLINDT, ET AL.
GROUP ART UNIT:	2172
EXAMINER:	CAM Y. T. TRUONG
TITLE:	"COPYING A PORTION OF A DATABASE STRUCTURE ASSOCIATED WITH A QUERY"
ATTORNEY DOCKET NUMBER:	069092.0118

INCLUDED IN THIS MAILING FOR THE ABOVE-REFERENCED PATENT APPLICATION ARE:

1. APPELLANTS' BRIEF (37 C.F.R. § 1.192) (IN TRIPLICATE);
2. FEE TRANSMITTAL (PTO/SB/17) WITH DUPLICATE COPY FOR FEE PROCESSING;
3. CHECK NO. 893525 IN THE AMOUNT OF $330.00 FOR FILING BRIEF IN SUPPORT OF APPEAL; AND
4. RETURN POSTCARD TO ACKNOWLEDGE RECEIPT OF ABOVE ITEMS.

ATTORNEY CONTACT: RONALD L. CHICHESTER REG. NO. 36,765
BAKER BOTTS L.L.P. PTO ID# 023640
TELEPHONE: 713.229.1341
FACSIMILE: 713.229.7741
EMAIL: RONALD.CHICHESTER@BAKERBOTTS.COM
PTO CUSTOMER NUMBER:

023640

PTO/SB/17 (10-03)
Approved for use through 07/31/2006. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

FEE TRANSMITTAL for FY 2004

Effective 10/01/2003. Patent fees are subject to annual revision.

☐ Applicant claims small entity status. See 37 CFR 1.27

TOTAL AMOUNT OF PAYMENT ($) 330

Complete if Known	
Application Number	069092.0118
Filing Date	June 27, 2001
First Named Inventor	Klindt, et al.
Examiner Name	Cam Y. T. Truong
Art Unit	2172
Attorney Docket No.	069092.0118

METHOD OF PAYMENT *(check all that apply)*

☑ Check ☐ Credit card ☐ Money Order ☐ Other ☐ None

☑ Deposit Account:

Deposit Account Number: 02-0383

Deposit Account Name:

The Director is authorized to: *(check all that apply)*

☐ Charge fee(s) indicated below ☑ Credit any overpayments

☑ Charge any additional fee(s) or any underpayment of fee(s)

☐ Charge fee(s) indicated below, **except for the filing fee** to the above-identified deposit account.

FEE CALCULATION

1. BASIC FILING FEE

Large Entity Fee Code	Large Entity Fee ($)	Small Entity Fee Code	Small Entity Fee ($)	Fee Description	Fee Paid
1001	770	2001	385	Utility filing fee	
1002	340	2002	170	Design filing fee	
1003	530	2003	265	Plant filing fee	
1004	770	2004	385	Reissue filing fee	
1005	160	2005	80	Provisional filing fee	

SUBTOTAL (1) ($)

2. EXTRA CLAIM FEES FOR UTILITY AND REISSUE

	Extra Claims		Fee from below	Fee Paid
Total Claims		-20** =		X =
Independent Claims		- 3** =		X =
Multiple Dependent				=

Large Entity Fee Code	Large Entity Fee ($)	Small Entity Fee Code	Small Entity Fee ($)	Fee Description
1202	18	2202	9	Claims in excess of 20
1201	86	2201	43	Independent claims in excess of 3
1203	290	2203	145	Multiple dependent claim, if not paid
1204	86	2204	43	** Reissue independent claims over original patent
1205	18	2205	9	** Reissue claims in excess of 20 and over original patent

SUBTOTAL (2) ($)

**or number previously paid, if greater; For Reissues, see above

FEE CALCULATION (continued)

3. ADDITIONAL FEES

Large Entity Fee Code	Large Entity Fee ($)	Small Entity Fee Code	Small Entity Fee ($)	Fee Description	Fee Paid
1051	130	2051	65	Surcharge - late filing fee or oath	
1052	50	2052	25	Surcharge - late provisional filing fee or cover sheet	
1053	130	1053	130	Non-English specification	
1812	2,520	1812	2,520	For filing a request for *ex parte* reexamination	
1804	920*	1804	920*	Requesting publication of SIR prior to Examiner action	
1805	1,840*	1805	1,840*	Requesting publication of SIR after Examiner action	
1251	110	2251	55	Extension for reply within first month	
1252	420	2252	210	Extension for reply within second month	
1253	950	2253	475	Extension for reply within third month	
1254	1,480	2254	740	Extension for reply within fourth month	
1255	2,010	2255	1,005	Extension for reply within fifth month	
1401	330	2401	165	Notice of Appeal	
1402	330	2402	165	Filing a brief in support of an appeal	330.00
1403	290	2403	145	Request for oral hearing	
1451	1,510	1451	1,510	Petition to institute a public use proceeding	
1452	110	2452	55	Petition to revive - unavoidable	
1453	1,330	2453	665	Petition to revive - unintentional	
1501	1,330	2501	665	Utility issue fee (or reissue)	
1502	480	2502	240	Design issue fee	
1503	640	2503	320	Plant issue fee	
1460	130	1460	130	Petitions to the Commissioner	
1807	50	1807	50	Processing fee under 37 CFR 1.17(q)	
1806	180	1806	180	Submission of Information Disclosure Stmt	
8021	40	8021	40	Recording each patent assignment per property (times number of properties)	
1809	770	2809	385	Filing a submission after final rejection (37 CFR 1.129(a))	
1810	770	2810	385	For each additional invention to be examined (37 CFR 1.129(b))	
1801	770	2801	385	Request for Continued Examination (RCE)	
1802	900	1802	900	Request for expedited examination of a design application	

Other fee (specify) _____

*Reduced by Basic Filing Fee Paid

SUBTOTAL (3) ($) 330.00

SUBMITTED BY

(Complete *(if applicable)*)

Name (Print/Type)	Ronald L. Chichester	*Registration No. (Attorney/Agent)*	36,765	*Telephone*	713.229.1341
Signature	Ronald L. Chichester			*Date*	July 7, 2004

WARNING: Information on this form may become public. Credit card information should not be included on this form. Provide credit card information and authorization on PTO-2038.

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:		Group Art Unit:	2172
	KLINDT, ET AL.		
Serial No.:	09/892,837	Examiner:	TRUONG, CAM Y.T.
Filed:	06/27/2001		
Title:	"COPYING A PORTION OF A DATABASE STRUCTURE ASSOCIATED WITH A QUERY"	Atty. Docket No.:	069092.0118

Honorable Commissioner for Patents
Washington, D.C. 20231

Attention: Board of Patent Appeals and Interferences

Dear Sir:

APPELLANTS' BRIEF (37 C.F.R. § 1.192)

This brief is submitted in support of Applicants' Notice of Appeal from the decision dated January 9, 2004 of the Examiner finally rejecting claims 1 through 30 of the subject application. The three-month shortened statutory period for filing the Notice of Appeal was due April 9, 2004. Applicants respectfully submitted a Notice of Appeal on May 7, 2004 with a Petition for One-Month Extension of Time. This brief is being filed within two months of the filing of the Notice of Appeal and is therefore considered timely filed.

This brief is transmitted in triplicate per 37 C.F.R. § 1.192.

07/13/2004 HALI11 00000014 09892837
01 FC:1402 330.00 OP

I. IDENTIFICATION OF THE REAL PARTY OF INTEREST

The real party in interest is:

NCR Corporation
1700 S. Patterson Blvd.
Dayton, OH 45479

by virtue of assignments by the inventors Jerry L. Klindt and Paul L. Sinclair.

II. IDENTIFICATION OF RELATED APPEALS AND INTERFERENCES

There are no related appeals or interferences.

III. STATUS OF ALL THE CLAIMS, PENDING OR CANCELLED, AND IDENTIFYING THE CLAIMS APPEALED

The application as originally filed contained 30 claims. All pending claims of the present application are reproduced in Appendix A, attached hereto.

IV. STATUS OF ANY AMENDMENT FILED SUBSEQUENT TO FINAL REJECTION

A response to the final office was filed by the Applicants on April 9, 2004. An Advisory Action was mailed by the examiner and was received by the Applicants on May 6, 2004. The Advisory Action indicated that the proposed amendments made in the response to the final office action would *not* be entered. Consequently, the status of the claims is as recited in Appendix A.

V. SUMMARY OF THE INVENTION

The present invention concerns databases, particularly relational databases that can perform multiple functions through use of appropriately formatted queries. The queries can be in the form of a structured query language ("SQL") query. Relational databases can contain one or more databases, and each database can contain one or more tables of information. Software

objects can be mapped to relational database tables using mapping techniques. Typically, the mapping technique employs an object definition that is kept within the database is used to generate the object from data stored in the database. More specifically, the present invention enables the copying of a portion of a database structure that contains one or more objects. The method of the present invention receives a query, such as an SQL query, and recursively retrieves the object definitions for the queried objects. The ultimate result is an ordered set of object definitions that were associated with the query. The ordered set of object definitions can be used to build a copy of a portion of the database structure on the same or a different computer. For example, the database structure can be used to create a second database (with different data) or the database structure could be populated with data from the original database, thereby making a copy of a portion of the original database.

The step of recursively retrieving object definitions can include a sub-step of recursively identifying objects that are associated with the query and categorizing them into a category. The step of recursively retrieving can be augmented by and retrieving the object definition for the categorized objects using a tool that corresponds to the category to which the object has been categorized. The categories can include tables, views, join indexes, triggers and macros. The tool may be, for example, a SHOW statement that is correlated to the category (i.e., SHOW TABLE when the cateogry is table, SHOW VIEW wne the category is view, and so on).

The method may further include the step of changing the ordered set of object definitions. The step of changing may include the reordering of the object definitions such that base objects are placed higher in the order so that those base objects exist when they are referenced by other objects. The method may further include the step of looking for references to the database

objects in a data dictionary while recursively retrieving the object definitions.

VI. CONCISE STATEMENT OF THE ISSUES PRESENTED FOR REVIEW

Claims 1, 2, 9-12, 19-22, 29-30 stand finally rejected under 35 U.S.C. 103(a) as being unpatentable over Bapat (U.S. Patent No. 5,295,256).

As to claims 1, 11 and 21, the examiner states that Bapat teaches the claimed limitations: "recursively retrieving object definitions for one or more database objects associated with a query to produce an ordered set of objects definitions" as (col. 10, lines 35-45). The examiner admitted "that Bapat does not clearly teach the claimed limitation 'building a copy of the database structure using the ordered set of object definition.'" Final Office Action, page 4. The examiner, however, states that Bapat teaches that header files are opened. The examiner further states that at step 176, the input from the files are read and control passes to step 178 where it is determined whether a class or a struct definition is recognized. The examiner states that if either of these is recognized, the struct or the class is recorded in the class hierarchy table and the class attribute table and that control then passes to step 184. If not, the examiner states, control passes to step 184 bypassing step 180. The examiner states that at step 184, it is determined whether or not a method definition has been recognized. The examiner states that if so, the method definition is recorded in the class method table and control passes to step 188, and if no method definition has been recognized at step 184, control passes directly to step 188 bypassing step 186. The examiner states that step 188 determines whether or not the end of the input has been reached. The examiner states that if no, [then] control passes back to step 176 where the next file is read. The examiner also states that class definitions are stored in order in the object class hierarchy and contended that each class definition is retrieved in order. The examiner states that the above information shows that the system builds a data structure based on retrieved

class or struct definitions by restoring retrieved class or struct definitions in class hierarchy table and class attribute table. Finally, the examiner states that class definitions, which include objects, are represented as object definitions (col. 9, lines 2-20). The examiner states that it would have been obvious to a person of an ordinary skill in the art at the time the invention was made to apply Bapat's teaching of restoring each retrieved class or struct definition in each file in class hierarchy table and class attribute table in order to allow any user can create a new data structure in order. However, Applicants do not believe that Bapat discloses, teaches or suggests the limitations of claims 1, 11 and 21 and, thus, does not render the claimed invention obvious. Applicants respectfully submit that claims 1, 11 and 21 are allowable over the prior art of record.

As to claims 2, 12 and 22, the examiner states that Bapat teaches the claimed limitations: "recursively identifying objects associated with the query (col. 10, lines 35-45); "categorizing each identified object into a category" as (col. 21, lines 30-60; col. 35, lines 65-67); "retrieving an object definition for each identified object using a tool corresponding to the category with which the identified object is connected" as each object in the hierarchy is processed by a loop beginning at step 292 which selects every class definition in the object class hierarchy. The examiner states that each class is retrieved and that the system has the type of object. The examiner states that the above information shows that the system has included a tool to retrieve an object definition corresponding to the type of an object are connected (col. 10, lines 35-45; col. 35, lines 65-67). Applicants do not believe that Bapat discloses, teaches or suggest the all of the limitations of claims 2, 12 and 22 and, thus, does not render the claimed invention anticipated nor obvious. Applicants respectfully submit that claims 2, 12, and 22 are allowable over the prior art of record.

As to claims 9, 19 and 29, the examiner states that Bapat teaches the claimed limitation

"the object definitions are ordered so that each object definition is ordered before the definition of any object that reference it" as each object in the hierarchy is processed by a loop beginning at step 292 which selects every class definition in the object class hierarchy. The examiner states that each class definition is retrieved. The examiner also states that multiple inheritance-in which a class may inherit attributes from more than on parent class is easily handled by creating one reference column as a pointer into the schema for each parent class. The examiner states that the above information shows that class definitions are stored in order in the object class hierarchy before the definition of any object that reference it (col. 10, lines 35-45; col. 23, lines 45-55). However, Applicants do not believe that Bapat discloses, teaches or suggests the limitations of claims 9, 19 and 29 and, thus, does not render the claimed invention anticipated nor obvious. Applicants respectfully submit that claims 9, 19 and 29 are allowable over the prior art of record.

As to claims 10, 20 and 30, the examiner states and Applicants agree that Bapat does not clearly teach the claimed limitation "recursively retrieving object definition for one or more database object includes looking for references to the one or more database objects in a data dictionary". However, the examiner states that Bapat teaches that each object in the hierarchy is processed by a loop beginning at step 292 which selects every class definition in the object class hierarchy. The examiner states that each class definition is retrieved. The examiner states that multiple inheritance-in which a class may inherit attributes from more than on parent class-is easily handled by creating one reference column as a pointer into the schema for each parent class. The examiner states that an Object Dictionary contains metaclass information, or information about the overall schema of the application domain. The examiner states that the population of the Object Dictionary was described in detail in connection with Figure 11 of

Bapat. The examiner states that the Object Dictionary contains the list of all classes, and includes information about attributes, superclasses, subclasses, and methods (col. 10, lines 35-45; col. 23, lines 45-55; Col. 44, lines 45-60). The examiner states that since objects are stored in hierarchy which included classes, thus, it is obvious that retrieving object definition includes looking for references to the one or more database objects in a object dictionary. The examiner states that it would have been obvious to a person of an ordinary skill in the art at the time the invention was made to apply Bapat's teaching of retrieving each class definition in the object class hierarchy. The examiner states further that the object definition contains list of all classes, superclasses, subclasses, that multiple inheritance-in which a class may inherit attributes from more than on parent class is easily handled by creating one reference column as a pointer into the schema for each parent class in order to read or create structure of objects during processing objects. However, Applicants do not believe that Bapat discloses, teaches or suggests the limitations of claims 10, 20 and 30 and, thus, does not render the claimed invention anticipated nor obvious. Applicants respectfully submit that claims 10, 20 and 30 are allowable over the prior art of record.

Claims 3, 13 and 23 stand finally rejected under 35 U.S.C. 103(a) as being unpatentable over Bapat (U.S. Patent No. 5,295,256) in view of Nackman et al (or hereinafter "Nackman") (U.S. Patent No. 6,182,281).

As to claims 3, 13 and 23, the examiner states that Bapat teaches the claimed limitations the categories include tables and views, join indexes, trigger" as tables, views, the join column into the parent class table will be used as the unique index, triggers (col. 8, line 20, col. 37, lines 30-35; col. 40, lines 1-5). Applicants agree with the examiner that Bapat fails to teach the claimed limitation "macros". However, the examiner states that Bapat teaches the different type

of objects including tables, views, joined indexes (col. 8, line 20, col. 37, lines 30-35; col. 40, lines 1-5). Furthermore, the examiner states that Nackman teaches Macros (col. 7, lines 35-37). The examiner states that it would have been obvious to a person of an ordinary skill in the art at the time the invention was made to apply Nackman's teaching of macros to Bapat's system in order to store a object in a dictionary. However, Applicants do not believe that Bapat or Nackman, either independently or in combination, disclose, teach or suggest the limitations of claims 3, 13 and 23 and, thus, does not render the claimed invention anticipated nor obvious. Applicants respectfully submit that claims 3, 13 and 23 are allowable over the prior art of record.

Claims 4, 14 and 24 stand finally rejected under 35 U.S.C. 103(a) as being unpatentable over Bapat in view of Nackman and Tung Ng et al (or hereinafter "Tung") (U.S. Patent No. 6,279,008).

As to claims 4, 14 and 24, the examiner states that Bapat discloses the claimed limitation subject matter in claim 1, except the claimed limitation "the tool is view statement if the identified object.... as a macro". The Applicants agree with the examiner that Bapat does not disclose the limitation "the tool is view statement if the identified object ... is a macro." The examiner states that Bapat teaches the source SQL statement for defining triggers, create table SQL command (col. 10, lines 40-45; col. 40, lines 1-5). Also, the examiner states that Tung's teaching of show-table-view button 1105 to show a view of tables corresponding to the database state 1204. The examiner states that the table view permits access to tables and database information associated with the database application state 1206 (col. 11, lines 60-67). The examiner states that Nackman teaches any Macros defined in the source, recognized by #defined command (col. 10, lines 32-35). The examiner states that it would have been obvious to a person of an ordinary skill in the art at the time the invention was made to apply Bapat's teaching of the

source SQL statement for defining triggers, create table SQL command, Tung's teaching of show-table-view button 1105 to show a view of tables corresponding to the database state 1204. The examiner further states that the table view permits access to tables and database information associated with the database application state 1206 and Nackman's teaching of any Macros recognized by # defined command in order to allow any user can have many choices for displaying objects or displaying configuration of any object to a user. However, Applicants do not believe that Bapat, Tung or Nackman, either independently or in combination, disclose, teach or suggest the limitations of claims 4, 14 and 24 and, thus, does not render the claimed invention anticipated nor obvious. Applicants respectfully submit that claims 4, 14 and 24 are allowable over the prior art of record.

Claims 5, 15 and 25 stand finally rejected under 35 U.S.C. 103(a) as being unpatentable over Bapat in view of Ma et al (or hereinafter "Ma") (U.S. Patent No. 5,920,725).

As to claims 5, 15 and 25, the examiner states that Bapat teaches the claimed limitation "retrieving unretrieved object definition...the query" as (col. 10, lines 35-45). The examiner states, and Applicants agree, that Bapat does not teach the claimed limitation "adding to the set of objects known to be associated with query...associated with the query...repeating items a and b...associated with the query". The examiner states that Bapat teaches each object in the hierarchy is processed by a loop beginning at step 292 which selects every class definition in the object class hierarchy. The examiner further states that each class definition is retrieved (col. 10, lines 35-45). Also, the examiner states that Ma teaches to insert a new field in database records requires that the database's format or schema be modified, step 30. The examiner states that adding the cell-phone held to the database's records can be accomplished with the statement: alter table employee add cellno varchar 20, which alters the employee table by adding a field

named "cellno" having up to 20 characters. The examiner states that the interfaces or input and output parameters for program objects which read database records are modified, step 32. The examiner states that the interfaces of many objects can be modified by changing the data structure for accessing the database by adding the new field: Class employee [private: char name[64]; char address[255]; char officeno[20]; char hiredate[l0]; float salary; char dept[32]; (col. 2, lines 50-67). The examiner states that the above information shows that the system add a field named cellno associated with query to a data structure which include class employee. Finally, the examiner states that this class employee is represented as a object definition. The examiner states that it would have been obvious to a person of an ordinary skill in the art at the time the invention was made to apply Bapat's teaching of retrieving object definitions and Ma's teaching adding a field name to class employee in order to maintain object definition. However, Applicants do not believe that Bapat or Ma, either independently or in combination, disclose, teach or suggest the limitations of claims 5, 15 and 25 and, thus, does not render the claimed invention anticipated nor obvious. Applicants respectfully submit that claims 5, 15 and 25 are allowable over the prior art of record.

Claims 6, 16 and 26 stand finally rejected under 35 U.S.C. 103(a) as being unpatentable over Bapat in view of Tung.

As to claims 6, 16 and 26, the examiner states that Bapat discloses the claimed limitation subject matter in claim 1, except the claimed limitation "sending the ordered set of object definitions from a first computer to a second computer". The examiner also states that Bapat teaches retrieving each object definitions (col. 10, lines 35-45). Also, the examiner states that Tung teaches that client sends database requests over Internet to server (col. 5, lines 55-57). The examiner states that it would have been obvious to a person of an ordinary skill in the art at the

time the invention was made to apply Bapat's teaching of retrieving each object definitions and Tung's teaching of sending database requests from client computer to server in order to allow a user can create a new database structure based on retrieved object definitions. However, Applicants do not believe that Bapat or Tung, either independently or in combination, disclose, teach or suggest the limitations of claims 6, 16 and 26 and, thus, does not render the claimed invention anticipated nor obvious. Applicants respectfully submit that claims 6, 16 and 26 are allowable over the prior art of record.

Claims 7-8, 8-18, 27-28 stand finally rejected under 35 U.S.C. 103(a) as being unpatentable over Bapat in view of Henckel (U.S. Patent No. 6,105,036).

As to claims 7, 17 and 27, the examiner states that Bapat discloses the claimed limitation subject matter in claim 1, except the claimed limitation "changing the order of the ordered set of object definitions". The examiner states that Bapat teach[es] each object in the hierarchy is processed by a loop beginning at step 292 which selects every class definition in the object class hierarchy. The examiner states that each class definition is retrieved. The examiner further states that this information shows that class definitions are stored in order in the object class hierarchy. The examiner states each class definitions is retrieved in order (col. 10, lines 35-45). The examiner states Henckel teaches that the ordered arrangement of object definitions such that a visual indication of the arrangement of such object definitions in source code file is maintained. The examiner states that it means that the ordered arrangement of object definitions is modified (abstract, col. 6, lines 20-50). The examiner states that it would have been obvious to a person of an ordinary skill in the art at the time the invention was made to apply Bapat's teaching of retrieving each object definitions and Henckel's teaching of the ordered arrangement of object definitions such that a visual indication of the arrangement of such object definitions in source

code file is maintained to Bapat's system in order to save time for searching or displaying a object. However, Applicants do not believe that Bapat or Henckel, either independently or in combination, disclose, teach or suggest the limitations of claims 7, 17 and 27 and, thus, does not render the claimed invention anticipated nor obvious. Applicants respectfully submit that claims 7, 17 and 27 are allowable over the prior art of record.

As to claims 8, 18, and 28, the examiner states that Bapat discloses the claimed limitation subject matter in claim 1, except the claimed limitation "changing the order of the order set of object definition.....reference it". Applicants agree that Bapat does not contain the aforementioned limitation. The examiner states that Bapat teach [sic] each object in the hierarchy is processed by a loop beginning at step 292 which selects every class definition in the object class hierarchy. The examiner states that each class definition is retrieved. The examiner states that this information shows that class definitions are stored in order in the object class hierarchy. Thus, each class definitions is retrieved in order (col. 10, lines 35-45). The examiner states that Henckel teaches that the ordered arrangement of object definitions such that a visual indication of the arrangement of such object definitions in source code file is maintained. The examiner states that it means that the ordered arrangement of object definitions is modified (abstract, Col. 6, lines 20-50). The examiner states that it would have been obvious to a person of an ordinary skill in the art at the time the invention was made to apply Bapat's teaching of retrieving each object definitions and Henckel's teaching of the ordered arrangement of object definitions such that a visual indication of the arrangement of such object definitions in source code file is maintained to Bapat's system in order to save time for searching or displaying a object. However, Applicants do not believe that Bapat or Henckel, either independently or in combination, disclose, teach or suggest the limitations of claims 8, 18 and 28 and, thus, does not

render the claimed invention anticipated nor obvious. Applicants respectfully submit that claims 8, 18 and 28 are allowable over the prior art of record.

VII. GROUPING OF CLAIMS

Claims 1 to 10 stand or fall together, claims 11 to 20 stand or fall together, and claims 21 to 30 stand and fall together.

VIII. ARGUMENTS OF THE APPELLANTS, WITH EACH ISSUE IN SEPARATE HEADINGS, WITH RESPECT TO EACH ISSUE PRESENTED FOR REVIEW

Applicants respectfully submit that the examiner has applied Bapat incorrectly in that the reference merely discloses the translation of *all* objects *outside* of a database *into* a database in the form of a database schema. Moreover, the Bapat reference merely reviews the objects to be copied systematically rather than recursively. Bapat's description does not need recursion because of the nature of the problem that Bapat was trying to solve. In stark contrast, the present invention starts with information from *within* the database and copies *a portion* of the structure (that already exists within the database) into another database, or even into another computer.

With respect to the independent claims:

A. Bapat does not teach or suggest copying a portion of a portion of a database structure by recursively retrieving object definitions in response to a query as claimed in claims 1, 11, and 21.

The Examiner states in her response to Arguments (page 2, Final Office Action, Paper No. 7) made in the previous Office Action Response that:

> Applicant's arguments filed 10/29/03 have been fully considered but they are not persuasive. Claims 1-30 are pending in this Office Action.
>
> Applicant argues that Bapat does not teach "each object in the hierarchy is associated with a query and recursively retrieving object definitions for one or more database objects associated with

a query to produce an ordered set of object definition." Bapat teaches that each object in the hierarchy is processed by a loop beginning at step 292, which selects every class definition in the object class hierarchy. Each class definition is retrieved at step 292 until no more objects in hierarchy are processed. Each object in the hierarchy is processed when the application make a call to the new method such as retrieve. It means that objects in the hierarchy are associated with the application's call. The application's call is represented as a query. Recursively is defined as to repeat again. Since each class definition, which is retrieved in a loop, is repeated many times until no more objects class in hierarchy is processed, thus, each class definition in hierarchy is retrieved recursively and in order. Each class definition contains one or more object definitions. For the above reasons, definitely, when system retrieves each class definition in hierarchy, the system retrieves each object definition of the class definition to produce an ordered set of object definitions too (fig. 13, col. 10, lines 35-65; col. 6, lines 65-67).

Applicant argues that Bapat does not teach "building a copy of the database structure using the ordered set of object definitions." Bapat teaches that a class hierarchy structure is translated into the relational table structure by mapping parent class 22 to a table 32. In a similar manner, class 24 is mapped to a table 38. A class in hierarchy includes objects definitions. Since, the system maps parent class 22 to a table 32 before mapping class 24 to a table 38, thus the system retrieves set of class definitions or object definitions in order. By mapping a class hierarchy structure into the relational table, the system creates a relational table structure, which contains all of classes or objects of the hierarchy structure. Thus, the relational table structures are presented as a copy of the hierarchy class structure. The hierarchy class structure is a data structure (col. 6, lines 60-65; col. 7, lines 1-5, figs. 1-3).

Applicant argues that Bapat does not teach "categorizing each identified object into a category." Bapat teaches class site contains object name, address, phone number, site-category. Class vendor contain object vendor name, object vendor address, object vendor phone number. The above information shows that each object is classified into different classes. Each class is represented as a category (col. 19, lines 35-50).

For the above reason, examiner believed that rejection of the last office action was proper.

Applicants' arguments filed on 10/29/2003 are reproduced below:

Applicants respectfully disagree and submit that Bapat does not disclose, teach, or suggest each and every feature recited in Applicants' independent Claims 1, 11, and 21. For example, Bapat does not disclose teach or suggest "recursively retrieving object definitions for one or more database objects associated with a query to produce an ordered set of object definitions," as recited in Applicants' Claim 1. To the contrary, Bapat merely discloses "a translator which converts object-oriented representations of data into relational tables." (Column 5, lines 33-36). For example, Bapat provides that a class construct may include a parent class 22 having "several attributes associated with it," a derived class 24 that "inherits the attributes of the patent class 22 and . . . has its own attributes," and a derived class 26 that "inherits attributes from derived class 24 and parent class 22 and contributes its own attributes." (Column 6, lines 34-49). Bapat provides "a mechanism for mapping this hierarchical schema into a relational table schema." (Column 6, lines 55-57). "[The] parent class 22 is mapped to a table 32 which is named after the class name (P1), . . . a table 38 is constructed representing derived class 24, . . . [and] a third table 44 is constructed to represent derived class 26." (Column 6, line 62 through Column 7, line 9). Accordingly, Bapat is limited to a system for imposing structure on a hierarchical schema to produce tables representing each class in the hierarchical schema. As such, although Bapat discloses that "each object in the hierarchy is processed by a loop beginning at step 292 which selects every class definition in the object class hierarchy," the class definitions are merely used with a "CREATE TABLE" SQL command to construct a table with the same table name. (Column 10, lines 38-45). Thus, Bapat does not teach that each object in the hierarchy is associated with a query as required by Claim 1. For at least these reasons, Applicants respectfully submit that Bapat does not disclose "recursively retrieving object definitions for one or more database objects associated with a query to produce an ordered set of object definitions," as recited in Applicants' Claim 1.

Additionally, Applicants submit that Bapat does not disclose, teach, or suggest, "building a copy of the database structure using the ordered set of object definitions," as recited in Applicants' Claim 1. The Examiner has acknowledged that the recited features and operation are absent from the teachings of Bapat but states that "it would have been obvious to a person of ordinary skill in the art at the time the invention was made to apply Bapat's teaching of restoring each retrieved class or struct definition in each file in class hierarchy table and class attribute table in order to allow any user to create a new data structure in

order." (Office Action, page 3). Applicants respectfully disagree. "To support the conclusion that the claimed invention is directed to obvious subject matter, either the references must expressly or impliedly suggest the claimed invention or the Examiner must present a convincing line of reasoning as to why the art is one that would have found the claim to be obvious in light of the teachings of the references." M.P.E.P. § 706.02(j) (citing *Ex parte Clapp*, 227 U.S.P.Q. 972, 973 (Bd. Pat. App. & Inter. 1985)). The Examiner presents neither. Not only does the Examiner fail to cite any support for his conclusion, but the Examiner's conclusion does not follow from the disclosure of Bapat. As discussed above, Bapat relates to a system for translating data from a hierarchy scheme to a relational scheme for storage purposes. As such, the system of Bapat uses object definitions for a class to impose structure on a hierarchical schema to produce tables representing each class in the hierarchical schema. If a "proposed modification or combination of the prior art would change the principle of operation of the prior art invention being modified, then the teachings of the references are not sufficient to render the claims *prima facie* obvious." M.P.E.P. § 2143.01. Applicants respectfully submit that the modification of Bapat, as suggested by the Examiner, would "change the principle of operation" of Bapat. Accordingly, Bapat also does not disclose, teach, or suggest "building a copy of the database structure using the ordered set of object definitions," as recited in Applicants' independent Claim 1.

Independent Claims 11 and 21 recite certain features and operations that are similar to the features and operations discussed above. For example, Claim 11 recites a computer-readable medium operable to "recursively retrieve object definitions for one or more database objects associated with a query to produce an ordered set of object definitions" and "build a copy of the database structure using the ordered set of object definitions." Claim 21 recites "recursively retrieving object definitions for one or more database objects associated with a query to produce an ordered set of object definitions." Thus, for reasons similar to those discussed above with regard to Claim 1, Applicants respectfully submit that Bapat does not disclose, teach, or suggest each and every feature and operation as set forth in Applicants' Claims 11 and 21.

For these reasons, Applicants respectfully request that the rejections of Claims 1, 11 and 21 be withdrawn.

The Applicants reiterate the above argument with the same force and effect as originally

submitted. The Applicants further note that Bapat, upon which the rejection of the independent claims stand or fall, is directed to mapping objects into relational database tables. Bapat does not teach or suggest mapping *some* of the objects, but rather *all* of the objects. In contrast, the present invention obtains information on only a portion of the database. Moreover, Bapat does not teach or suggest a *recursive* retrieval of objects from *within* the database, but rather a systematic generation of relational database representations that are put *into* the Bapat database. Bapat does not teach or suggest recursive retrieval of object definitions from within the database because Bapat is directed to another purpose -- one that is unrelated to retrieving object definitions from within the database. That is why Bapat does not disclose, teach, or suggest the limitations of independent claims 1, 11 and 21, and that is why Bapat does not render independent claims 1, 11 and 21 obvious under 35 U.S.C. §103(a). M.P.E.P. §706.02(j), *Ex parte Clapp, supra*. For these reasons, Applicants respectfully request that the rejections of Claims 1, 11 and 21 be withdrawn.

Claims 2-10 depend from claim 1, claims 12-20 depend from claim 11, and claims 22-30 depend from claim 21, and contain all limitations thereof. Therefore the dependent claims should be allowable for the same reasons as independent claims 1, 11 and 21. Applicants respectfully submit that the cited reference does not disclosed an apparatus, system or method which anticipates the present invention as claimed in independent claims 1, 11 and 21, or in the claims that depend upon claims 1, 11 and 21. Therefore, it is respectfully submitted that all claims are allowable over the prior art of record, and such allowance is earnestly solicited.

Summary

In light of the foregoing, Applicants respectfully request that the final rejection of the pending claims should be reversed and the application be remanded for allowance of the pending claims, or, alternatively, remand the application for further examination if appropriate references can be found by the examiner.

Applicants submit herewith a check for the fee for filing a brief in support of an appeal in the amount of $330.00. Applicants believe that there are no additional fees due in association with the filing of this Appeal Brief. However, should the Commissioner deem any additional fees as being due, including any fees for any additional extensions of time, the Commissioner is requested to accept this as a Petition therefore, and is hereby authorized to charge any additional fees due, or to credit any overpayment, to Baker Botts L.L.P. Deposit Account No. **02-0383**, Order Number **069092.0118** under 37 C.F.R. § 1.16 or § 1.17.

Respectfully submitted,

BAKER BOTTS L.L.P. (023640)

By: Ronald L. Chichester

Ronald L. Chichester
Registration No. 36,765
Baker Botts L.L.P.
One Shell Plaza
910 Louisiana Street
Houston, Texas 77002-4995
Telephone: 713.229.1341
Facsimile: 713.229.7741
EMail: Ronald.Chichester@bakerbotts.com
ATTORNEY FOR APPLICANTS

Date: July 7, 2004

APPENDIX A

(original) A method for copying a portion of a database structure, the database including one or more database objects, the method comprising

recursively retrieving object definitions for one or more database objects associated with a query to produce an ordered set of object definitions;

building a copy of the database structure using the ordered set of object definitions.

2. (original) The method of claim 1, where recursively retrieving object definitions includes

recursively identifying objects associated with the query;

categorizing each identified object into a category;

retrieving an object definition for each identified object using a tool corresponding to the category with which the identified object is connected.

3. (original) The method of claim 2, where the categories include tables, views, join indexes, triggers and macros.

4. (original) The method of claim 2, where the tool is

a. a SHOW VIEW statement if the identified object is categorized as a view;

b. a SHOW TABLE statement if the identified object is categorized as a table;

c. a SHOW JOIN INDEX statement if the identified object is categorized as a join index;

d. a SHOW TRIGGER statement if the identified object is categorized as a trigger;

e. a SHOW MACRO statement if the identified object is categorized as a macro.

5. (original) The method of claim 1, where recursively retrieving object definitions includes

a. retrieving unretrieved object definitions for a set of objects known to be associated with the query;

b. adding to the set of objects known to be associated with the query those objects contained in the retrieved object definitions that are not already in the set of objects known to be associated with the query;

c. repeating items a and b until no new objects are added to the set of objects known to be associated with the query.

6. (original) The method of claim 1, further comprising sending the ordered set of object definitions from a first computer to a second computer.

7. (original) The method of claim 1, further comprising changing the order of the ordered set of object definitions.

8. (original) The method of claim 7, where changing the order of the ordered set of object definitions includes reordering the object definitions so that each object definition is ordered before the definition of any object that references it.

9. (original) The method of claim 1, where the object definitions are ordered so that each object definition is ordered before the definition of any object that references it.

10. (original) The method of claim 1, where recursively retrieving object definitions for one or more database objects includes looking for references to the one or more database objects in a data dictionary.

11. (original) A computer-readable medium containing computer-executable code for instructing a computer to:

recursively retrieve object definitions for one or more database objects associated with a query to produce an ordered set of object definitions;

build a copy of the database structure using the ordered set of object definitions.

12. (original) The computer-executable code of claim 11, in which, when recursively retrieving object definitions, the computer:

recursively identifies objects associated with the query;

categorizes each identified object into a category;

retrieves an object definition for each identified object using a tool corresponding to the category with which the identified object is connected.

13. (original) The computer-executable code of claim 12, where the categories include tables, views, join indexes, triggers and macros.

14. (original) The computer-executable code of claim 12, where the tool is

a. a SHOW VIEW statement if the identified object is categorized as a view;

b. a SHOW TABLE statement if the identified object is categorized as a table;

c. a SHOW JOIN INDEX statement if the identified object is categorized as a join index;

d. a SHOW TRIGGER statement if the identified object is categorized as a trigger;

e. a SHOW MACRO statement if the identified object is categorized as a macro.

15. (original) The computer-executable code of claim 11, in which, when recursively retrieving object definitions, the computer:

a. retrieves unretrieved object definitions for a set of objects known to be associated with the query;

b. adds to the set of objects known to be associated with the query those objects contained in the retrieved object definitions that are not already in the set of objects known to be associated with the query;

c. repeats items a and b until no new objects are added to the set of objects known to be associated with the query.

16. (original) The computer-executable code of claim 11, further comprising computer-executable code instructing the computer to send the object definitions from a first computer to a second computer.

17. (original) The method of claim 11, further comprising computer-executable code instructing the computer to change the order of the ordered set of object definitions.

18. (original) The computer-executable code of claim 17, in which, when changing the order of the ordered set of object definitions, the computer reorders the object definitions so that each object definition is ordered before the definition of any object that references it.

19. (original) The computer-executable code of claim 11, in which, when storing the definitions, the computer stores the definitions so that each object definition is ordered before the definition of any object that references it.

20. (original) The computer executable code of claim 11, in which, when recursively retrieving object definitions for one or more database objects, the computer looks for references to the one or more database objects in a data dictionary.

21. (original) A package of data useful in building a copy of a database structure generated in accordance with the following act:

recursively retrieving object definitions for one or more database objects associated with a query to produce an ordered set of object definitions.

22. (original) The package of data of claim 21, where the object definitions are recursively retrieved in accordance with the following acts:

recursively identifying objects associated with the query;

categorizing each identified object into a category;

retrieving an object definition for each identified object using a tool corresponding to the category with which the identified object is connected.

23. (original) The package of data of claim 22, where the categories include tables, views, join indexes, triggers and macros.

24. (original) The package of data of claim 22, where the tool is

a. a SHOW VIEW statement if the identified object is categorized as a view;

b. a SHOW TABLE statement if the identified object is categorized as a table;

c. a SHOW JOIN INDEX statement if the identified object is categorized as a join index;

d. a SHOW TRIGGER statement if the identified object is categorized as a trigger;

e. a SHOW MACRO statement if the identified object is categorized as a macro.

25. (original) The package of data of claim 21, where the object definitions are recursively retrieved in accordance with the following acts:

a. retrieving unretrieved object definitions for a set of objects known to be associated with the query;

b. adding to the set of objects known to be associated with the query those objects contained in the retrieved object definitions that are not already in the set of objects known to be associated with the query;

c. repeating items a and b until no new objects are added to the set of objects known to be associated with the query.

26. (original) The package of data of claim 21, further comprising the act of sending the ordered set of object definitions from a first computer to a second computer.

27. (original) The package of data of claim 21, further comprising the act of instructing the computer to change the order of the ordered set of object definitions.

28. (original) The package of data of claim 27, where the act of changing the order of the ordered set of object definitions includes reordering the object definitions so that each object is ordered before the definition of any object that references it.

29. (original) The package of data of claim 21, where the act of storing object definitions includes storing the object definitions so that each object definition is ordered before the definition of any object that references it.

30. (original) The package of data of claim 21, where the act of recursively retrieving object definitions for one or more database objects includes looking for references to the one or more database objects in a data dictionary.